



Implementació de la tecnologia IoT en una aplicació bàsica de smart home

Pablo Sust de Rosselló

Memòria del projecte

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 3 of 35		

REVISION HISTORY AND APPROVAL RECORD

Revision	Date	Purpose
0	6/06/2019	Document creation

DOCUMENT DISTRIBUTION LIST

Name	E-mail
[Pablo Sust de Rosselló]	
[Albert Aguasca Sole]	

WRITTEN BY:		REVIEWED AND APPROVED BY:	
Date	6-6-2019	Date	6-6-2019
Name	Pablo Sust de Rosselló	Name	Albert Aguasca
Position	Project author	Position	Project Supervisor

0. CONTINGUTS

0.	ContInguts	4
1.	Introducció	5
1.1.	Objectiu.....	5
1.2.	Justificació	5
1.3.	Abast	6
1.4.	Especificacions bàsiques	6
2.	Antecedents.....	7
2.1.	Benchmarking	7
2.2.	Gestió de l'energia en la llar	8
2.3.	Internet of Things (IoT).....	8
2.4.	Smart home	9
2.5.	Sistemes de gestió de l'energia	9
2.6.	Noves tecnologies.....	11
3.	Estudis comparatius.....	12
3.1.	PLCs, Gateways i Microcontroladors	12
3.2.	Plataforma IoT	13
3.3.	Protocols de comunicació	15
4.	Cas d'aplicació.....	16
4.1.	Introducció	16
4.2.	Dispositius de camp.....	17
4.3.	Configuració del PLC	18
4.4.	Missatges MQTT.....	21
4.5.	Configuració del servidor	23
5.	Pressupost.....	31
5.1.	Despeses del projecte.....	31
5.2.	Cost del prototip.....	31
5.3.	Cost del servei	32
6.	Treball futur.....	33
6.1.	Dispositius de la llar	33
6.2.	Gestió de l'energia	33
6.3.	Benchmarking	33
6.4.	Automatització de la millora	33
7.	Conclusions	34
8.	Bibliografia.....	35

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 5 of 35		

1. INTRODUCCIÓ

1.1. *Objectiu*

Aquest treball és un primer pas dins un projecte més ambiciós que pretén ser una plataforma per a compartir dades de consum energètic.

El públic al que va destinat el projecte és el consumidor final o, de forma més general, les associacions d'usuaris o inclús empreses que aglutinin consumidors finals, i la finalitat és obtenir les “millor pràctiques” que ens ensenyin com cal gestionar l'energia dins la nostra llar.

Aquest primer pas al que ens referim consisteix en implementar una aplicació de la tecnologia IoT (Internet of Things) en un entorn domèstic que permeti, sobretot, la fàcil escalabilitat a diverses llars en la mateixa estructura de control centralitzat

1.2. *Justificació*

Tan en un present, com cada cop amb més importància en un futur, l'estalvi d'energia en la llar ha de ser un objectiu principal de la domòtica.

Els estàndards d'homologació tant en els electrodomèstics com, més recentment, en els propis habitatges, representen un gran avanç. Però encara hi ha molta feina a fer en els hàbits d'ús, el seguiment i el control de l'estalvi energètic.

Les estratègies proposades fins ara es limiten a uns consells que entren dins el sentit comú, i algunes recomanacions d'associacions sense ànim de lucre o de companyies de subministrament d'energia. Però no se'n fa un seguiment.

És ja un clàssic la frase “No es pot gestionar el que no es pot mesurar”, i creiem que el màxim aprenentatge i la gestió més òptima es pot produir quan les mesures, a més d'individualitzades es poden treballar de forma conjunta i comparativa. També és possible emprar, en un futur ben proper, i si la base és prou ampla, la tècnica del benchmarking per a la millor continuïtat de la gestió energètica a les llars.

Aquesta necessitat és la que ha impulsat la realització d'aquest projecte que pretén establir la base tecnològica sobre la que implementar un projecte de gestió energètica més ambiciós i que contempli la millora de forma col·lectiva.

Per altra banda, l'evolució de l'anomenada Internet of Things (IoT) en la Indústria 4.0 i l'aparició en el mercat de d'una gran varietat d'equips i serveis fan que un projecte així quedi a l'abast de gairebé tothom.

I demostrar-ne la viabilitat és la motivació subjacent a tot el treball.

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 6 of 35		

1.3. *Abast*

Els principals objectius que es pretenen assolir en aquesta aplicació són:

- 1.- Capacitat d'actuar manualment sobre un element de la llar des d'un equip mòbil o de la web
2. Actuar de forma automàtica des d'un Sistema central sobre un actuador de la llar
3. Ubicar sensors en una llar que puguin enviar les mesures a l'estructura central
4. Dissenyar i construir una base de dades que porti un registre tan de les actuacions automàtiques com de les manuals i del conjunt de mesures preses pels diferents sensors de la llar

1.4. *Especificacions bàsiques*

La implementació d'aquest projecte requerirà dels següents components:

- Una llar amb accés a internet i un router WiFi
- Un servidor dedicat o un proveïdor de serveis de "cloud" on ubicar-hi l'estructura centralitzada
- Un software adient a cada un dels mòduls de l'estructura

Els requeriments per al treball en si mateix no són gaires. Durant la descripció dels diferents components es mirarà de donar una idea dels requeriments que calgui per a escalar-lo

Aquest projecte pretén únicament demostrar la viabilitat de la idea, i per tant ha de tenir les especificacions mínimes per a assolir-ho, és a dir::

- Un Microcontrolador a la llar que:
 - o Llegeixi les mesures d'un sensor i les envii al servidor
 - o Activi i desactivi un dispositiu seguint ordres del servidor o autònomament segons un algorisme que ho estableixi
- Una plataforma de missatgeria per a distribuir els missatges
- Un servidor on s'hi pugui executar la lògica centralitzada del sistema
- Un servidor web que permeti al usuari interactuar amb el sistema
- Una base de dades amb capacitat per enregistrar tots els esdeveniments

Per a la finalitat actual no hi ha prestacions mínimes doncs es tracta únicament de demostrar la viabilitat tècnica, però cal fer-ho amb eines que permetin una important escalabilitat

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 7 of 35		

2. ANTECEDENTS

2.1. *Benchmarking*

El Benchmarking és una tècnica de gestió empresarial que normalment s'adscriu dins les tècniques de millora continua o de qualitat total (TQM, Total Quality Management), però que es pot emprar de forma independent, i de fet així es va crear.

És la tècnica de gestió que la companyia Xerox va començar a treballar al 1979 quan estava en una crisi molt forta i de la que se'n va sortir gràcies a la seva aplicació.

De totes formes no va començar a ser popular en el món dels negocis fins a partir de 1989 que el Dr. Robert C. Camps va publicar el llibre "Benchmarking, The Search for industry Best Practics which Lead to Superior Performance".

Sense entrar en detalls sobre la metodologia ni la tècnica en sí, que per altra banda ha tingut moltes versions i evolucions, cal destacar-ne els principis bàsics, car són els que han inspirat i donen sentit a aquest treball.

El funcionament bàsic de cada etapa consisteix en les següents fases:

- Selecciona el producte, servei, procés o departament d'una empresa que es vol millorar.
- Seleccionar del mercat, qui són els que ho fan millor.
- Determinar les mètriques que s'usaran per a determinar i mesurar tan el que es vol millorar com els paràmetres que poden influir en els resultats.
- Mirar d'esbrinar el que fa que una organització fa "millor" que les altres en el sentit determinat pels indicadors, i obtenir-ne les "millors pràctiques".
- S'implementen les millors pràctiques.

Un procés de Benchmarking és un procés de millora continua, així que aquestes fases es van repetint cíclicament de forma indefinida.

Cal destacar que tant l'àmbit d'actuació, com la selecció dels millors, com les mètriques son components dinàmics del Benchmarking.

En aquest projecte no es pretén definir mètriques ni grups, però és necessari tenir-ho en compte pel disseny de l'estructura de dades i la missatgeria on es tracta la informació.

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 8 of 35		

2.2. *Gestió de l'energia en la llar*

Normalment la gestió d'energia en la llar ha consistit tradicionalment en estalviar-la, simplement apagar llums, baixar temperatura del termòstat i aïllar la casa a base de persianes tancades. Però actualment hi ha un conjunt força ampli d'elements que, si bé permeten optimitzar el consum sense sacrificar confort, en compliquen la gestió:

- a) Les tarifes variables segons horari i els descomptes i tarifes diferents dels diversos operadors.
- b) Els múltiples aparells programables i automatismes possibles en la llar.
- c) La generació de la pròpia energia amb panells solars.

També hi ha la possibilitat d'adquirir habitatges i electrodomèstics en diferents qualitats d'eficiència energètica, però això porta a preguntar-se si val la pena econòmicament o fins a quin punt realment hi ha estalvi energètic.

2.3. *Internet of Things (IoT)*

La internet de les coses en anglès, és un concepte que es refereix a la interconnexió digital dels objectes a través d'internet sense la participació directa de les persones.

A diferència de que en el seu temps es va denominar M2M ("Machine to Machine"), on la connexió era entre ordinadors i/o màquines industrials complexes, les "coses" a les que es refereix IoT són objectes que poden arribar a ser molt quotidians, com interruptors, termòmetres, persianes, electrodomèstics, etc.

En la pràctica els objectes es comuniquen directament amb un PLC (Programmable Logic Controller) i aquest a través d'un Gateway és qui envia i rep informació a través d'internet a "el núvol" on hi haurà els serveis que gestionaran els objectes.

De totes formes, en entorns que no requereixen la robustesa de les instal·lacions de producció industrial, l'aparició dels microcontroladors a preus molt més baixos que els del PLC o els Gateway han obert les portes a la popularització.

Així, en un projecte on el hardware de connexió d'un sensor o un actuator a internet que pot costar més de 2.000€ i ser rentable en indústria, ara es pot fer per menys de 10€ i aplicar-ho a qualsevol llar.

A més, a preus realment econòmics hi podem trobar al mercat tota mena de sensors com termòmetres, amperímetres, fotòmetres, i actuadors com relés, leds i alarmes sonores, tots ells preparats per ser gestionats per un microcontrolador.

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 9 of 35		

2.4. *Smart home*

I amb la IoT i la intel·ligència artificial, el antic concepte d'automatització de la llar anomenat domòtica ha evolucionat al nou Smart Home o "cases intel·ligents", que tenen una forta intersecció amb aquest projecte.

La aparició de les grans empreses Google, Amazon i Apple en aquest negoci ha fet que en poc temps hi hagi gran quantitat d'aparells connectables i gestionables per IoT, no obstant això, els seus protocols són tancats i la seva orientació no és ni compartir dades excepte que sigui amb el fabricant ni a l'estalvi energètic, sinó a l'oci i al confort.

2.5. *Sistemes de gestió de l'energia*

Apart de les Smart Home, hi ha empreses i productes que tenen enfoc orientats directa o indirectament a l'estalvi energètic, però que per motius diversos que detallarem en cada cas, no s'han pogut aprofitar realment com antecedents per a aquest projecte però sí que mereixen ser esmentats com a plantejaments propers que ja són al mercat, i dels que en un futur se n'hauria de plantejar com a competidors o col·laboradors o ambdues coses a la vegada

N'hi ha diversos però se n'han seleccionat quatre per ser quatre models diferents que completen la visió del que s'està fent en el mercat

2.5.1. PickData



És una empresa de Viladecavalls amb vocació internacional, el seu producte és molt complet, tan en equips, de fabricació pròpia, com en el software i els serveis que ofereix. És una empresa orientada a aconseguir que el seu client gestioni millor l'energia i disposa d'una

gran varietat d'indicadors inclús de gràfiques comparatives, però el seu enfoc no està en el compartir per millorar i el seu client objectiu és la indústria.

Els equips són massa cars per a una implantació a una llar particular i el producte és tancat.

Es podria plantejar en un futur una col·laboració a nivell de fabricació d'equips i de coneixement ja que en cap cas la idea va dirigida al mateix destinatari, però de totes formes la tecnologia que fan servir per als seus clients tampoc seria la ideal per a les llars particulars per una qüestió de preu.

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 10 of 35		

2.5.2. Efergy



Una empresa amb seu al Regne Unit però amb implantació de més d'un milió de llars en 50 països.

El seu objectiu també és l'estalvi energètic i comercialitza equips i el software de gestió. Tan els equips com el software estan orientats a

consumidor final i tenen un aspecte molt atractiu, donant la imatge de senzillesa i comoditat.

Es tracta d'un enfoc de mesurar per tenir el control, i en això es basa l'estalvi, no tenen ni vocació de compartir ni disposen de protocols oberts com per a que les seves solucions es puguin aprofitar per aquest projecte.

De totes formes d'aquesta empresa cal aprendre'n la imatge de senzillesa que donen i la importància de l'estètica del producte

2.5.3. Wattio



És una empresa de Donostia, els seu producte té un enfoc molt més clar cap al confort i la seguretat que a la llar que cap a l'estalvi energètic en si mateix, no obstant això, també hi posen l'accent en l'estalvi i ho fan com les altres,

basat en mesura i el control com a mostra, a la seva web hi ha la frase "Cuanto mejor conoces tu consumo, más ahorras".

El seu producte és la domòtica més que el Smart Home i es presenten com "El sistema domótico más valorado", així que compleixen l'estàndard de IFTTT, que els hi permet la compatibilitat amb Google Home, però no ha servit pel projecte, doncs la seva tecnologia s'aparta del IoT

2.5.4. OpenEnergyMonitor



És una empresa de Eryri, a Snowdonia, al nord del País de Gales amb l'enfoc més similar al d'aquest projecte.

El seu objectiu és l'estalvi energètic i el seu públic objectiu és el consumidor final a la llar, i com els altres, no ho fan per la via de la compartició.

Disposen de tres línies de productes:

- Monitorització del consum d'energia
- Monitorització de la generació per plaques solars
- Monitorització de temperatura i humitat

Una gran diferència amb aquest projecte és que ells no contemplen els actuadors, és a dir, encara que aporten informació pel control i la optimització, el seu sistema no dóna la possibilitat d'automatitzar la gestió, només monitoritzar

Fabriquen els seus equips i el seu software, i ho fan tot en codi obert i públic, usant microcontroladors RaspberryPi i comunicació amb MQTT

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 11 of 35		

El problema és que al ser una tecnologia molt madura i que contempla moltes variants, la complexitat d'intentar aprofitar-ne alguna part era molt superior a la de plantejar un projecte més senzill i començar des del principi

2.6. *Noves tecnologies*

Convé destacar, en aquest capítol, un conjunt de noves tecnologies que es relacionen directament amb el projecte. Tot i que no té sentit aplicar-les en aquesta fase inicial, han estat necessàries per poder-lo plantejar dins un context d'evolució

- a) **Big Data** Prenent mostres cada 10 segons s'obtenen 8640 registres per cada sensor durant un dia, i 3.153.600 registres a l'any. Quan ens interressi creuar dades i obtenir indicadors de moltes llars amb diversos sensors a cada llar caldrà l'ús de tecnologies que permetin gestionar-ho.
- b) **IA, Xarxes neuronals, Machine Learning i Deep Learning** Si bé en un principi hauran de ser les persones les que analitzin les dades, en fases posteriors, quan, a més, les millors pràctiques es puguin implementar amb sistemes automàtics, l'optimització i l'anàlisi de les dades hauran de ser suportades per processos matemàtics més adients, ràpids i eficaços.
- c) **Cibersecurity**. Tractant-se d'informació sobre hàbits de vida de les persones i especialment sobre la presència dins la llar o no, per atraure a públic objectiu a grups de millora caldrà ser rigorós amb la seguretat.
- d) **Blockchain**. En la societat de la informació en la que estem immersos, cada cop es fa més evident la lluita de les grans corporacions per apoderar-se de tota mena de dades i més en concret de dades sobre els nostres hàbits de vida. No seria descartable que en un futur es volgués prescindir del proveïdor de serveis IaaS o PaaS i s'organitzés una comptabilitat descentralitzada garantint l'anonimat dels participants.

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 12 of 35		

3. ESTUDIS COMPARATIUS

3.1. *PLCs, Gateways i Microcontroladors*

Pel que fa al tipus de control sobre la llar, hi ha dos grans grups de tecnologia que es poden plantejar: PLC (Programmable Logic Controller) que és el que es fa servir majoritàriament en la indústria i amb la que ha nascut la Indústria 4.0, o els Microcontroladors, havent-se fet molt popular l'Arduino.

Els PLC són equips programables de gran robustesa i fiabilitat però de preus molt elevats en comparació amb els microcontroladors, i no seria viable en un entorn domèstic. A més, els PLC no tenen la possibilitat de comunicar-se amb internet i necessiten un equip igualment robust i car anomenat Gateway per a poder implementar IoT. Així l'elecció ha hagut de ser l'ús de Microcontroladors basats en Arduino o similars.

Arduino és una plataforma de hardware i software amb llicència de codi obert que consta de tres parts: La placa controladora (hardware), el software que s'executa en la placa, basat en Wiring, i l'entorn de desenvolupament IDE basat en Processing. Arduino té la capacitat de comunicar-se directament a través de WiFi sempre que s'hi connecti el mòdul especialitzat, i per tant, no necessita Gateway.

Arduino no és la única ni, tal vegada la millor en el sentit de prestacions tècniques, però per l'objectiu del projecte el criteri ha hagut de ser:

- Que disposi d'extensa documentació especialment del software, però també sobre la placa en sí i la seva connectivitat.
- Que el seu cost sigui molt baix. Es pretén que en la llar n'hi hagi diversos, i un preu alt faria inviable el projecte.
- Que tingui el mòdul WiFi integrat. Aquest requeriment no és necessari però facilita.

Així que dins la limitació de que hi ha d'haver molta documentació, el criteri a seguir ha estat bàsicament el preu, i com que es tracta de llicència de codi obert, s'ha optat per una placa "clònica" amb WiFi integrada de preu inferior als 6€ comprada individualment.

La placa triada ha estat la Lolin V3 amb WiFi integrat:



En el capítol següent se'n fa la descripció tècnica

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 13 of 35		

3.2. *Plataforma IoT*

En aquest punt ens centrarem en l'estudi de les principals plataformes software i hardware que hi ha actualment al mercat. Totes elles ofereixen un ecosistema IoT, es a dir, que la plataforma ens oferirà software i hardware i tot el necessari pel desenvolupament de l'aplicació.

3.2.1. El núvol: Conceptes IAAS, PAAS, SAAS

Quan no està clara la dimensió que pot arribar a agafar el projecte o si necessitem gestionar recursos de forma variable, avui en dia convé contractar el servidor com a servei en "el núvol", és a dir, sense importar ni quin és el servidor o servidors físics ni on són ni totes les tasques relacionades amb el seu manteniment.

Però quan ens referim a contractar el servidor com a servei, cal especificar quin tipus de servei, doncs existeixen diverses modalitats en funció del nivell de delegació i preus que ens pugui interessar. D'entre les diferents alternatives tenim:

- e) **SaaS** (Software as a Service). És el més antic, popularitzat en les grans empreses abans que es popularitzés el nom de "el núvol", però que darrerament està aconseguint els majors èxits. Es tracta del nivell màxim de delegació, on el que contracta només n'és un usuari (administrador) i les tasques de manteniment correctiu, evolutiu, còpies de seguretat, configuracions, ajustos, rendiment, seguretat, etc. Són responsabilitat del proveïdor. Exemples clàssics en són la botiga d'internet, la comptabilitat i el gestor de correu electrònic. És la modalitat més cara però la que requereix menys esforç si ens podem ajustar a l'oferta concreta de servei que dona el proveïdor.
- f) **Paas** (Plataform as a Service): És el paradigma dels desenvolupadors, on només cal programar en els llenguatges que suporti la plataforma i al proveïdor se li delega la resta. Un exemple n'és Google App Engine que primer només permetia Ruby, però ara ja s'ha ampliat a Java, Node.js, Python i altres. En aquestes plataformes és el proveïdor qui s'encarrega del desplegament.
- g) **Iaas** (Infrastructure as a Service): És la modalitat amb menys nivell de delegació però al mateix temps la que permet més flexibilitat. En aquesta plataforma podem seleccionar el sistema operatiu i instal·lar les aplicacions que es desitgi. És la modalitat de menys cost i la més adaptable, però la que requereix més responsabilitat del client i més risc.

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 14 of 35		

3.2.2. El mercat

Hi ha un número molt gran del proveïdors en cada una de les modalitats. Per les necessitats del projecte hem seleccionat uns quants proveïdors a estudiar segons els següents criteris:

- a) Que ofereixin la modalitat PaaS
- b) Que permetin un gran nivell d'escalabilitat
- c) Que ofereixin garanties de fiabilitat, creixement i continuïtat
- d) Que ofereixin Node.js i html per a desenvolupament
- e) Que ofereixin Microsoft SQL Server

Els dos darrers requeriments venen donats pel coneixement del que es disposa per fer el desenvolupament

Així hem mirat els següents proveïdors

- a) **Microsoft Azure**
- b) **Google Cloud Platform**
- c) **Amazon Web Services**
- d) **IBM Bluemix**

Tots ells disposen d'ofertes des de SaaS fins a IaaS en diverses especialitzacions.

Hem descartat Microsoft Azure pels alts preus d'entrada, que en les fases inicials del projecte es fan inviables, així com hem descartat Google Cloud i IBM Bluemix pel mateix motiu. Tot i que aquestes darreres ofereixen \$300 i \$200 respectivament en serveis de Cloud, en cap d'elles hi hem trobat serveis gratuït per a volums baixos.

Amazon Web Services, malgrat tenir una estructura de preus de moltes variables, ofereix serveis gratuïts que es converteixen en serveis de pagament automàticament quan els volums creixen.

3.2.3. La elecció

Per motius de preu, s'ha triat AWS (Amazon Web Services).

En concret s'ha contractat un servei EC2, és a dir, un servidor virtual en modalitat IaaS, on s'hi ha instal·lat el front-end i el back-end. S'ha contractat també el servei RDS on s'hi ha obert una instància de MS SQL Server.

El motiu de triar EC2 en modalitat IaaS és que així, en el servidor virtual, s'hi ha pogut instal·lar "Mosquito", el software que gestiona MQTT

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 15 of 35		

3.3. *Protocols de comunicació*

Atenent al grau d'implantació, es podria dir que només hi ha dos protocols de comunicació entre "coses" per a la implementació de IoT:

- a) **OPC UA** (Open Platform Communications), que abans es deia OLE (Object Linking and Embedding) for Process Control. Basat en una tecnologia de Microsoft DCOM de comunicació de i entre objectes (lògics) en arquitectura SOA

Es tracta de tenir un servidor com a font de dades i qualsevol aplicació basada en OPC hi pot accedir per a llegir o escriure dades

Les sigles UA (Unified Architecture) es van afegir al obrir el protocol a qualsevol plataforma, es va adaptar a compartir objectes entre màquines distants i es va obrir el codi.

Actualment és un protocol obert i gratuït, amb llicència GPL 2.0

- b) **MQTT** (Message Queuing Telemetry Transport) és un estàndard ISO (ISO/IEC PRF 20922) basat en publicació i subscripció de missatges Treballa sobre TCP/IP. La primera versió és de 1999.

És un sistema de missatgeria molt eficient, dissenyat per a comunicacions lleugeres i amples de banda estrets, amb comunicacions asíncrones, desacoblades en temps i espai.

És un protocol obert, amb forces servidors (brokers) gratuïts i software gratuït per a crear-se un mateix el servidor.

És el més implantat fora de l'entorn de les grans fàbriques i disposa de molta documentació i software desenvolupat.

Per simplicitat, preu, existència d'ajudes i de software, s'ha triat MQTT

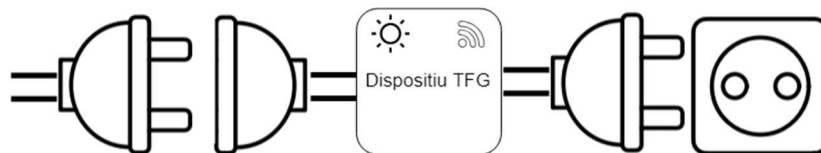
A més s'ha triat Mosquitto, el software que implementa un broker de MQTT més implantat i gratuït que s'ha instal·lat en el servidor EC2 contractar a Amazon

4. CAS D'APLICACIÓ

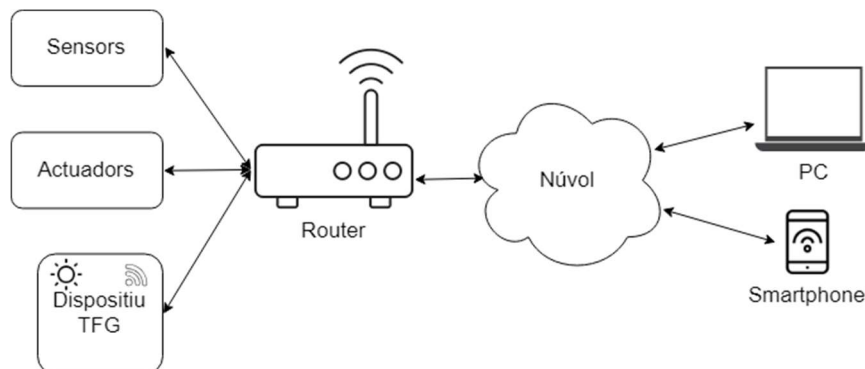
4.1. Introducció

El projecte consisteix en dissenyar un sistema capaç de gestionar l'energia d'un conjunt gran d'habitatges, obtenint informació rellevant de sensors, actuant sobre diversos actuadors i informant al usuari a l'hora que permetent-lo actuar també sobre la seva instal·lació des d'internet.

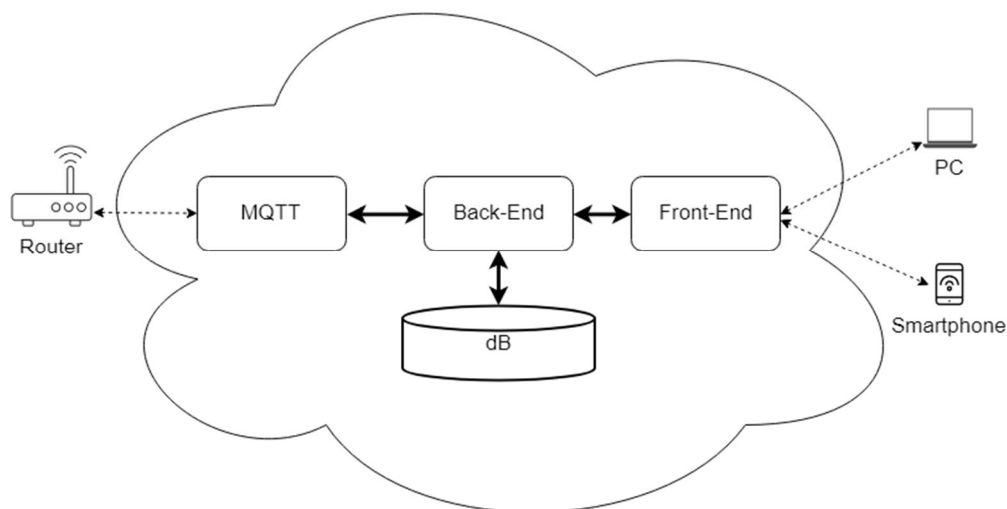
Com a demostració s'ha construït un dispositiu que intercepta el corrent elèctric entre el subministrament (endoll) i el consum (electrodomèstic), en mesura el consum, en permet la interrupció del subministrament, i té un pilot LED sobre el que també es pot actuar.



La següent figura representa l'esquema general del sistema:



I dins el núvol:



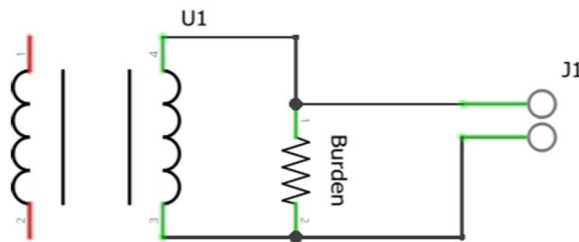
4.2. Dispositius de camp

En el prototipus de dispositiu s'hi han inclòs: Un sensor de corrent, un relé i un LED.

4.2.1. Sensor de corrent

S'ha emprat un sensor SCT-013-000, molt comú en el mercat.

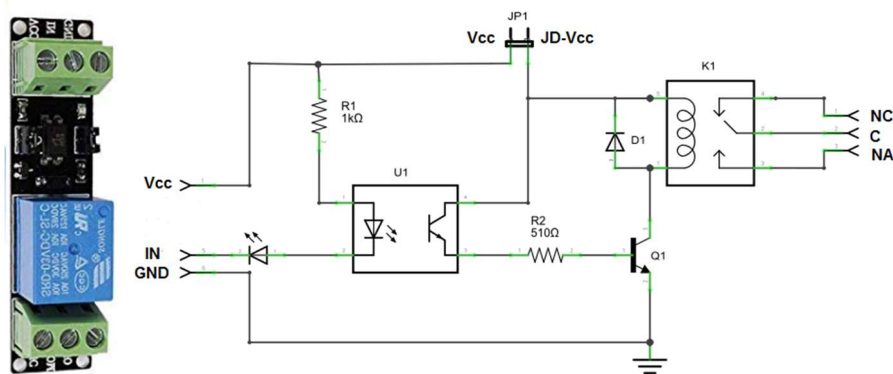
És un sensor de corrent alterna no invasiu, de nucli dividit, i amb capacitat de mesurar fins a 100A. El seu esquema elèctric és:



La sortida és un voltatge de corrent alterna i aproximadament de 1v per cada Ampere. I posat que a la placa s'ha de convertir a un corrent altern sempre positiu i amb una amplitud màxima de pic a pic de 3v, primer s'ha dividit el voltatge de la sortida de SCT013 i després agafat de referència $\frac{V_{max}}{2}$. En el esquema general es pot veure com ha quedat configurat.

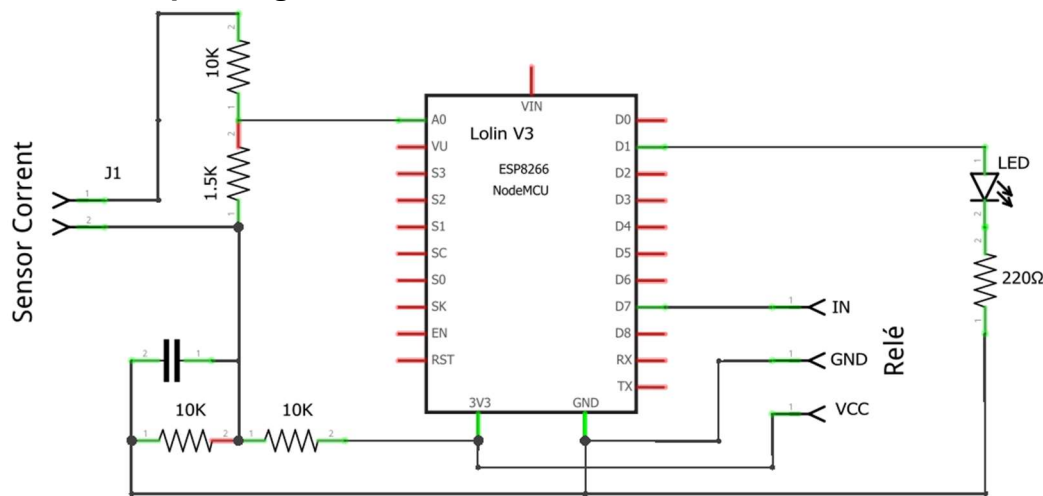
4.2.2. Relé

El Relé és un model molt comú que té com a característiques a destacar: Que s'activa amb 3v (quan el normal són 5v), i que funciona amb optoacoblador.



De l'esquema cal observar que el relé està actiu quan a la connexió IN hi ha un "0".

4.2.3. Esquema general



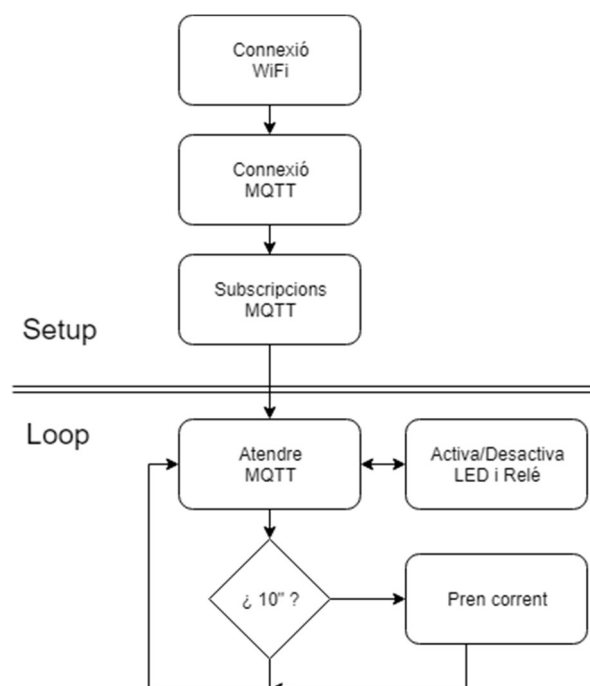
4.3. Configuració del Microcontrolador

El software del Microcontrolador és el d'Arduino i s'ha fet amb l'IDE d'Arduino.

El programa d'un microcontrolador Arduino té tres parts:

- Declaracions.** No s'executa codi.
- Setup.** S'executa una i només una vegada quan s'encén el microcontrolador.
- Loop.** És el programa que s'executa indefinidament.

Al programar cal anar en compte que es tracta d'un processador mono-tasca i el loop ha de donar servei a tot, per tant cal alliberar al procés el màxim per a que pugui donar el màxim de iteracions i així atendre puntualment tots els serveis. Aquest és l'esquema de funcionament del prototip:



Pel control del Loop es declara la variable:

```
const long interval = 10000; // interval de presa de mesura [ms]
```

El codi del Loop és el següent:

```
void loop() {  
  //double Irms = energyMonitor.calcIrms(1484);  
  client.loop();  
  unsigned long currentMillis = millis();  
  if (currentMillis - previousMillis >= interval) {  
    previousMillis = currentMillis;  
    //Pren corrent  
  }  
}
```

Client.loop s'explica a MQTT, i el codi complet està en els annexes.

4.3.1. MQTT

Per al funcionament de la missatgeria MQTT cal declarar la llibreria corresponent, identificar el servidor i el password i declarar la connexió amb la WiFi. També es declara la funció que atendra els missatges MQTT.

```
#include <PubSubClient.h> // Llibreria  
  
PubSubClient client(espClient); //Connexió amb WiFi  
  
// Dades de connexió al servidor MQTT  
const char* mqttServer = "ec2-3-16-125-122.us-east-2.compute.amazonaws.com";  
const int mqttPort = 1883;  
  
void callback(char* topic, byte* payload, unsigned int length)  
{..}
```

En el Setup es connecta, es subscriu als topics i s'enllaça la funció callback.

```
// MQTT  
client.setServer(mqttServer, mqttPort);  
client.setCallback(callback);  
  
while (!client.connected()) {  
  Serial.println("Connecting to MQTT...");  
  if (client.connect("ESP8266Client")) {  
    // if (client.connect("ESP8266Client", mqttUser, mqttPassword)) {  
  } {  
    Serial.println("Connected");  
  } else {  
    Serial.print("Failed with state ");  
    Serial.println(client.state());  
    delay(2000);  
  }  
}  
  
client.subscribe("08591MON87/MF001/OnOff");  
client.subscribe("08591MON87/MF001/Alarma");
```

I en el Loop crida la funció que mira si hi ha missatges, i si n'hi ha, es crida a `callback` per a que els atengui.

```
client.loop();
```

4.3.2. Dispositius de camp

L'entrada A0 és un conversor A/D que transforma el voltatge d'entrada de 0 a 3,3V a valors de 0 a 1023. Com que la mitja del corrent és inestable, es comença estimant 512 però després es va recalculant amb filtre autoregressiu de paràmetres 2 i 1. Calculat el valor, es multiplica per un factor que s'ha trobat empíricament i que es desa a la variable `FactorConversio`.

```
// Corrent
float Mitja = 512; // Aprox 1/2 del recorregut, però s'ajusta
float FactorConversio=6.21;
```

La funció de càlcul va prenent les mostres cada milisegon durant un segon i ho fa cada deu segons

```
float prenCorrent()
{
    float mesura;
    float sum = 0;
    float sum2 = 0;
    long temps = millis();
    int cont = 0;

    while (millis() - temps < 1000)
    {
        mesura = analogRead(A0);
        sum += mesura;
        sum2 += sq(mesura-Mitja);
        cont += 1;
        delay(1);
    }
    Mitja = (2*Mitja+sum/cont)/3;
    return(sqrt(sum2/cont)*FactorConversio);
}
```

I en el mateix loop es pren el corrent i es publica a MQTT:

```
//Pren corrent
digitalWrite(pinLedIntern,LOW);
String strPotMsg="{\"consum\": \" + String(prenCorrent()) +
}\"";
client.publish("08591MON87/MF001/Consum", (char*)
strPotMsg.c_str());
digitalWrite(pinLedIntern,HIGH);
```

4.3.3. WiFi

Declaració:

```
#include <ESP8266WiFi.h>

//WiFi
const int NumSsidOk=2;
```

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 21 of 35		

```
const char* ssid[NumSsidOk] ={"WiFiLocal","WiFiMobil"};
const char* password[NumSsidOk] ={"PswLocal","PswMobil"};
WiFiClient espClient;
```

Tan a `ssid` com `password` normalment són una única variable, però per aquest prototip, i només pel prototip, s'ha usat un algorisme que permet fer proves amb la WiFi de la casa, però que si es vol fer una demo fora de la casa, es pugui usar el mòbil.

Queda fora de l'abast d'aquesta memòria explicar aquest codi, però es deixen els codis font en els annexes per a poder-ho mirar.

4.4. Missatges MQTT

4.4.1. Introducció

Com ja s'ha mencionat, MQTT és un protocol basat en publicació i subscripció. Hi ha un servidor (broker) que és l'encarregat de rebre, emmagatzemar i distribuir els missatges, i uns clients que són els que publiquen missatges i s'hi subscriuen, demanant al broquer que els hi comuniqui tots els missatges que pertanyin a un determinat conjunt anomenat topic (pronunciat com tòpic i que la traducció més ajustada podria ser "tema").

Així doncs, cada missatge va associat a un i només un topic, però les subscripcions es poden fer de diversos topics o conjunts de topics.

Es detallen aquí els quatre tipus principals de servei que podem sol·licitar a un broker. S'exposen els que s'han fet servir per a aquest TFG amb les simplificacions i paràmetres que s'han fet servir.

a) **connect**

És necessari per a poder publicar o subscriure's, al connectar amb el boker cal especificar un Identificador de client, tot i que els missatges no contenen l'autor ni es pot restringir els subscriptors pel seu Id.

Segons l'administrador del broker hagi decidit, al broker s'hi ha d'accedir amb usuari i password o pot ser obert a tothom. Si s'ha restringit per usuari i password caldrà especificar-ho al fer connect.

En tot cas, prèviament, s'haurà hagut d'especificar un nom o IP de servidor i un port TCP.

b) **Publish**

Simplement es posa un missatge en el broker amb un topic i un QoS (que s'explica més endavant). El missatge és un text lliure, però si es vol posar un conjunt de dades, és recomanable fer-ho amb un format que després a Node.js li sigui fàcil d'interpretar.

Per al projecte s'ha triat usar text pla "1" o "0" pel LED i el Relé, però una estructura d'un únic camp a les lectures, per si més endavant s'hi vol posar l'hora o paràmetres de fiabilitat. Així el missatge té la forma:

```
{"consum":34.03}
```

c) **Subscribe**

Quan un client es subscriu a un topic o conjunt de topics, cada cop que algun altre client publica un missatge d'un dels topics, el broker li fa arribar al subscrit.

d) **Unsubscribe**

Cancel·la una subscripció. No es fa servir finalment en el projecte, però si durant el desenvolupament, per fer proves.

4.4.2. Topic

El topic és una cadena de caràcters codificats en UTF-8. Un topic consisteix en un o més nivells de topic, i cada nivell va separat pel símbol "/". A més, es poden usar els caràcters comodí "+", que es pot usar per qualsevol valor d'un nivell, o "#" que es pot usar al final. En aquest projecte, amb base de dades, l'estructura ha de ser funcional únicament pel servei de missatgeria propi. L'estructura dissenyada és:

```
Servidor/Instalacio/Element/Funcio
```

En el prototip actual, ja que només es treballa amb un servidor, i no és evident que l'estructura de diversos servidors pugui aportar algun benefici, s'ha treballat amb els tres darrers nivells, i a modus d'exemple s'ha usat la instal·lació: "08581MON87", que podria ser codi postal, acrònim de carrer i número o qualsevol altre cosa, i d'elements se n'han usat dos: "MF001" i "MF002" que podrien voler dir Multifuncio 001 i 2, ja que es tracta d'un endoll multifunció, podria ser llum, cuina, etc. De funcions s'han usat: `OnOff` pel relé, `Alarma` pel LED i `consum` pel sensor.

4.4.3. QoS

QoS vol dir Quality of Service. Els missatges es poden enviar en tres nivells diferents de qualitat:

- a) **QoS=0**: Els missatges s'envien i no hi ha garantia que arribin.
- b) **QoS=1**: Es garantitza que el missatge es rep una vegada, però pot ser que ho rebi més d'una vegada.
- c) **QoS=2**: Es garantitza que es rep una i només una vegada.

En aquest projecte cal usar en els missatges de `consum` QoS=0, doncs realment no passa res si es perd algun missatge, però en les ordres d'encendre i apagar cal usar QoS=1, ja que ens hem d'assegurar que s'activa o desactiva un element però rebre el missatge repetit no té cap efecte negatiu ja que s'usa de missatge "0" o "1".

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 23 of 35		

4.5. *Servidor: Back-End, Front-End i Base de Dades*

Com s'ha comentat en apartats anteriors, l'aplicació corre a una màquina virtual de Amazon Web Services. Descarregant l'arxiu de connexió remota i amb les credencials corresponents, s'accedeix a la màquina virtual on s'hi ha instal·lat Node.js.

- Server name: ec2-3-16-125-122.us-east-2.compute.amazonaws.com
- IP: 3.16.125.122

En aquest cas, s'ha utilitzat una màquina molt senzilla amb Windows 2012 Server R2 Standar.

També s'ha instal·lat, en el servei RDS, un Microsoft SQL Server Express Edition.

- Server: dbpfg01.cwdwkkdlzdof.us-east-2.rds.amazonaws.com
- Database: dbpfg01

4.5.1. Back End

S'ha utilitzat Node.js com a llenguatge de programació. El nom de l'arxiu es "main.js" i esta guardat a la carpeta `server`.

A continuació es pot veure l'arxiu de configuració Package.json del projecte.

```
{
  "name": "tfg",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "nodemon server/main.js"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "@chenfengyuan/datepicker": "^1.0.2",
    "express": "^4.16.4",
    "jquery": "^3.3.1",
    "moment": "^2.23.0",
    "mqtt": "^2.18.8",
    "msnodesqlv8": "^0.6.11",
    "node-localstorage": "^1.3.1",
    "socket.io": "^2.1.1",
  },
  "devDependencies": {
    "nodemon": "^1.18.7"
  }
}
```

S'ha utilitzat `nodemon` per al desenvolupament, fent que es reiniciï automàticament el servidor quan hi ha canvis al back-end.

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 24 of 35		

A continuació hi ha un llistat dels paquets utilitzats:

- **DatePicker:** A la pantalla de informes, hi ha dos datepickers que permeten seleccionar les dates.
- **Express:** El framework web per a node.js, estableix a quin port es vol publicar la aplicació.
- **Jquery:** S'ha utilitzat el framework jquery per facilitar el desenvolupament de funcions.
- **Moment:** Serveix per formatejar les dates.
- **Mqtt:** Permet la comunicació amb el broker de MQTT.
- **msnodesqlv8:** Permet la connexió amb el SQL per tal de poder fer consultes e insertar dades.
- **LocalStorage:** Permet guardar informació localment en el dispositiu.
- **Socket.io:** Permet la comunicació entre back-end i front-end.

El back-end s'encarrega de les següents funcions:

4.5.1.1. Validar el Login d'usuari

Quan l'usuari es vol logar, el back-end rep la petició a través de un socket, i consulta a la base de dades si l'usuari i contrasenya son correctes. En cas de ser correctes, el back-end envia al front-end l'usuari logat.

El servidor rep la següent petició:

```
socket.on('credencials', function(data) { ...})
```

Executa una consulta al SQL:

```
"SELECT NomUsuari, Contrasenya, GrupUsuari FROM
[TFG01].[dbo].[Uusuaris] WHERE NomUsuari = '"+data.usuari + '"'
```

On la variable `data.usuari` és la variable rebuda a la petició.

Finalment si l'usuari i la contrasenya són correctes, emet via socket la informació de l'usuari logat:

```
socket.emit('usuariinfo', usuariinfo);
```

4.5.1.2. Comunicació amb la llar per MQTT

A través de la llibreria `mqtt`, el back-end funciona com a client MQTT.

```
var mqtt = require("mqtt");
var client = mqtt.connect("mqtt://localhost:1883");
```

Actualment el servidor és el Mosquitto que s'ha instal·lat en la mateixa màquina virtual i per això usa `localhost`.

Aquest client MQTT, es subscriu als temes del microcontrolador per tal de que, quan el broker rebí dades en aquests temes, les rebí el back-end, i podrà publicar missatges per a que siguin rebuts per la llar.


```
client.on("connect", function () { //Subscripcio a MQTT
    client.subscribe("08591MON87/+Consum");
    console.log("Connected to Broker MQTT");
});
```

Quan es rep un missatge a un dels temes que el client MQTT esta subscrit, s'executa la següent funció:

```
client.on("message", function (topic, message) {... }
```

Rebem del broker tan el topic com el missatge. Segons quin hagi sigut el topic, es fa una cosa o una altre.

Seguidament hi ha un exemple del que es fa quan es rep un missatge al topic de "MF001".

```
var data = JSON.parse(message);
if (topic == "08591MON87/MF001/Consum"){
    var DISPOSITIU01 = [{
        consum: data.consum
    }];
    DISPOSITIU_CONSUM[0] = { //Salva la dada en local
        consum: data.consum
    };
    io.emit('DISPOSITIU01', DISPOSITIU01); //Envia les dades
                                         al front-end

    sql.close();
    sql.connect(config, function (err) { //Salva a la base
        de dades  SQL
        if (err) console.log(err);
        var request = new sql.Request();
        request.query("INSERT INTO
[TFG01].[dbo].[Historics] ([DataHora], [Dispositiu],
[Consum]) VALUES ('"+ getTime() +
"', '08591MON87/MF001', '"+ data.consum+"')", function
(err, recordset) {
            if (err) console.log(err);
        });
    });
}
```

Cada cop que el client MQTT del back-end rep un missatge, aquest ho envia al front-end a través de un socket i ho guarda a la base de dades.

Des del front-end, quan l'usuari vol encendre o apagar el dispositiu, el back-end rep la petició 'on' o bé la 'off' que s'envia al back-end amb el corresponent socket:

```
//Activa el dispositiu
socket.on('on', function(data) {
    var options={qos:1};
    var topic = "08591MON87/MF00"+data.dispositiu+"/OnOff";
    client.publish(topic, "1",options); //Publica MQTT
});

//Desactiva el dispositiu
socket.on('off', function(data) {
    var options={qos:1};
    var topic = "08591MON87/MF00"+data.dispositiu+"/OnOff";
    client.publish(topic, "0",options); //Publica MQTT
});
```

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 26 of 35		

L'enviament es fa amb nivell de servei QoS=1 ja que no hi ha cap problema de que ho rebí per duplicat, però sí ens hem d'assegurar que ho rep.

4.5.1.3. Gestió automàtica (Alarmes)

Cada 2 segons es comprova si el valor dels diferents llindars que hi ha posats per a cada dispositiu multi-funció és menor que el valor del consum rebut.

```
setInterval(checkAlarms, 2000);
```

En cas de que sigui més alt, i l'alarma no estigués activa, es genera l'alarma i s'envia tant al front-end com al SQL.

Així, cada 2000 milisegons s'executa la funció:

```
function checkAlarms () {...}
```

Aquesta funció compara els valors de llindar que té memoritzats a la matriu DISPOSITIU_PARAM i els compara amb les darreres lectures de consum.

Si cal activar l'alarma: Ho registra a la base de dades, activa el senyal en el front-end i encen el LED del dispositiu.

```
io.emit('ALARMES', DISPOSITIU_ALARMES);
sql.close();
sql.connect(config, function (err) { //Guarda l'alarma al SQL
  if (err) console.log(err);
  var request = new sql.Request();
  request.query("INSERT INTO [TFG01].[dbo].[LogAlarmes]
([DataHoraOn], [Dispositiu], [Descripcio]) VALUES ('"+
getTime() +"', '08591MON87/MF00"+dispositiu+"', 'Alarma consum
alt dispositiu 0"+dispositiu+"')", function (err, recordset) {
    if (err) console.log(err);
  });
});
var topic = "08591MON87/MF00"+(i+1)+"/Alarma";
client.publish(topic, "1"); //Publica al broker de MQTT
```

Si cal desactivar-la, actualitza el registre d'alarmes amb la data de desactivació, desactiva el senyal en el front-end i apaga el LED del dispositiu:

```
io.emit('ALARMES', DISPOSITIU_ALARMES);
dispositiu = i+1;
sql.close();
sql.connect(config, function (err) { //Actualitza l'hora de
desactivació de l'alarma
  if (err) console.log(err);
  var request = new sql.Request();
  request.query("UPDATE [TFG01].[dbo].[LogAlarmes] set
DataHoraOff = '"+ getTime() +" where Id = (SELECT
MAX (Id) FROM [TFG01].[dbo].[LogAlarmes] where
Dispositiu = '08591MON87/MF00"+dispositiu+"')",
function (err, recordset) {
    if (err) console.log(err);
  });
});
var topic = "08591MON87/MF00"+(i+1)+"/Alarma";
```

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 27 of 35		

```
client.publish(topic, "0"); //Publica al broker de MQTT
```

Però aquesta gestió porta intrínseca una altra associada que és la de mantenir la base de dades i la memòria actualitzada als darrers valor que posa l'usuari en el front-end, i això ho fa amb el següent socket:

```
socket.on('Param_Change', function(data) {
    sql.close();
    sql.connect(config, function (err) { //Update the SP to
the SQL
        if (err) console.log(err);
        var request = new sql.Request();
        request.query("UPDATE [TFG01].[dbo].[Parametres] SET
Parametre = " + data.param + " WHERE Dispositiu =
'08591MON87/MF00"+ data.dispositiu +"'"; function (err,
recordset) {
            if (err) console.log(err);
        });
    });
    DISPOSITIU_PARAM[data.dispositiu - 1].param =
data.param;
});
```

4.5.1.4. Consultes a la base de dades per informes

Quan el front-end necessita consultar la base de dades per mostrar-ne la informació que hi ha, ho sol·licita al back-end a través d'un socket. En el projecte n'hi ha dos, un per Alarmes i l'altre pels històrics de consum ('getalarms' i 'gethist'). Es mostra el de 'getalarms'.

```
socket.on('getalarms', function() {
    var data = [];
    sql.close();
    sql.connect(config, function (err) { //Select last 50
alarms in SQL
        if (err) console.log(err);
        var request = new sql.Request();
        request.query("SELECT TOP 50 DataHoraOn, Dispositiu,
Descripcio, DataHoraOff FROM [TFG01].[dbo].[LogAlarmes] order
by Id desc", function (err, recordset) {
            data = recordset.recordset;
            for (i=0; i<data.length; i++){ //Date format
                data[i].DataHoraOn =
moment(data[i].DataHoraOn).format('DD/MM/YYYY HH:mm:ss');
                if (data[i].DataHoraOff!= null){
                    data[i].DataHoraOff =
moment(data[i].DataHoraOff).format('DD/MM/YYYY HH:mm:ss');
                }
                else{
                    data[i].DataHoraOff = "No ha retornat";
                }
            }
            socket.emit('historic-alarms', data);
        });
    });
});
```

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 28 of 35		

```
});
```

Quan el back-end obté la informació de la base de dades, li dóna al front-end mitjançant el socket 'historic-alarmes'.

La pàgina d'històric de consums és anàloga però té dos informes, es pot veure el codi complert en els annexes.

4.5.2. Front End

S'ha programat amb html, css i javasacript. S'ha utilitzat el framework javascript jquery, per tal de facilitar la programació amb javascript.

S'accedeix al back-end a través de la IP 3.16.125.122 per el port 80.

Primer de tot, apareix una pantalla per logar-se.



Al prémer "Log In" es consulta a la base de dades com ja s'ha explicat i es passa a la pàgina principal:

Dispositiu 01
 Descripció del dispositiu 01

ON OFF

 Consum actual:

27.15

 Consum màxim alarma:

1000

☐ Alarma consum alt

Dispositiu 02
 Descripció del dispositiu 02

ON OFF

 Consum actual:

0

 Consum màxim alarma:

800

☐ Alarma consum alt

Dispositiu 03
 Descripció del dispositiu 03

ON OFF

 Consum actual:

0

 Consum màxim alarma:

40

☐ Alarma consum alt

Dispositiu 04
 Descripció del dispositiu 04

ON OFF

 Consum actual:

0

 Consum màxim alarma:

40

☐ Alarma consum alt

Dispositiu 05
 Descripció del dispositiu 05

ON OFF

 Consum actual:

0

 Consum màxim alarma:

40

☐ Alarma consum alt

Dispositiu 06
 Descripció del dispositiu 06

ON OFF

 Consum actual:

0

 Consum màxim alarma:

30

☐ Alarma consum alt

© 2019 TFG Pau Sust

Log off

La pàgina està preparada per veure simultàniament fins a sis dispositius, però pel projecte només se n'ha desenvolupat dos.

Per a cada dispositiu hi ha el botó de connectar-lo i el de desconnectar-lo, així com un lloc on veure el consum de la darrera lectura i un indicador d'estat d'alarma que s'activa quan el consum està per sobre del nivell que s'ajusta en la casella superior.


En el menú superior hi ha la possibilitat de veure l'informe d'Alarmes o del històric del dispositiu.

L'informe d'alarmes és de les darreres alarmes que s'han activat, independentment del dispositiu:

Alarmes

Data Hora	Dispositiu	Descripció	Data Hora Retorn
20/06/2019 13:34:51	08591MON87/MF001	Alarma consum alt dispositiu 01	20/06/2019 13:35:01
20/06/2019 11:56:10	08591MON87/MF001	Alarma consum alt dispositiu 01	20/06/2019 11:56:40
19/06/2019 18:36:15	08591MON87/MF002	Alarma consum alt dispositiu 02	19/06/2019 18:36:25
19/06/2019 18:35:45	08591MON87/MF002	Alarma consum alt dispositiu 02	19/06/2019 18:35:55
19/06/2019 18:35:15	08591MON87/MF002	Alarma consum alt dispositiu 02	19/06/2019 18:35:25
19/06/2019 16:37:38	08591MON87/MF001	Alarma consum alt dispositiu 01	19/06/2019 16:37:48
19/06/2019 16:37:18	08591MON87/MF001	Alarma consum alt dispositiu 01	19/06/2019 16:37:28
19/06/2019 16:31:27	08591MON87/MF001	Alarma consum alt dispositiu 01	19/06/2019 16:31:37
18/06/2019 23:59:51	08591MON87/MF001	Alarma consum alt dispositiu 01	19/06/2019 00:00:02
18/06/2019 23:59:21	08591MON87/MF001	Alarma consum alt dispositiu 01	18/06/2019 23:59:35
18/06/2019 23:10:24	08591MON87/MF001	Alarma consum alt dispositiu 01	18/06/2019 23:10:54

I en l'informe històric, permet seleccionar dues dates i el dispositiu que es vol mirar i presentar les dades segons el criteri escollit.


UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

[INICI](#)
[ALARMES](#)
[INFORMES](#)

Informes

Data inicial:
 Data final:

Alarmes

Data Hora	Dispositiu	Descripció	Data Hora Retorn
20/06/2019 13:34:51	08591MON87/MF001	Alarma consum alt dispositiu 01	20/06/2019 13:35:01
20/06/2019 11:56:10	08591MON87/MF001	Alarma consum alt dispositiu 01	20/06/2019 11:56:40
19/06/2019 16:37:38	08591MON87/MF001	Alarma consum alt dispositiu 01	19/06/2019 16:37:48
19/06/2019 16:37:18	08591MON87/MF001	Alarma consum alt dispositiu 01	19/06/2019 16:37:28
19/06/2019 16:31:27	08591MON87/MF001	Alarma consum alt dispositiu 01	19/06/2019 16:31:37

Dades històriques

Data Hora	Dispositiu	Consum
20/06/2019 20:14:01	08591MON87/MF001	15.54
20/06/2019 20:13:51	08591MON87/MF001	15.69
20/06/2019 20:13:41	08591MON87/MF001	15.54
20/06/2019 20:13:31	08591MON87/MF001	16.03
20/06/2019 20:13:21	08591MON87/MF001	15.59
20/06/2019 20:13:11	08591MON87/MF001	15.26
20/06/2019 20:13:01	08591MON87/MF001	15.61
20/06/2019 20:12:51	08591MON87/MF001	15.51

4.5.3. SQL

S'ha creat una base de dades utilitzant Microsoft SQL Server 2016, instal·lat al servei RDS. S'ha creat l'usuari de SQL `sa`, que és des de el qual es connecta el back-end.

S'ha creat l'estructura mínima per a provar el funcionament del projecte que consisteixen en:

- Usuaris:** Les dades per poder accedir.
- Parametres:** On s'emmagatzema el llindar de cada dispositiu per a que s'activi l'alarma
- LogAlarmes:** Pel registre d'activacions i desactivacions de les alarmes.
- Historics:** Pel registre dels consums. Registrant consum i hora com es pot veure en el informe.

Els detalls del camps es troben en els annexes en forma d'instruccions "CREATE" de SQL.

5. PRESSUPOST

Dividim el pressupost en tres parts, la primera és la del propi projecte, la segona correspon a la llar i la tercera al sistema central. La primera serà un cost únic, la segona tindrà un cost aproximadament lineal amb el número de punts de control, i la tercera sol tenir un creixement logarítmic.

5.1. *Despeses del projecte*

S'inclouen les hores aproximades dedicades a cada part del projecte. El cost és aproximat, basat en preus de mercat consultat a professionals, comptant que el preu de mercat de l'autònom és la meitat del preu de facturació de les empreses d'enginyeria.

	Preu / hora (€/h)	Hores (h)	Preu (€)
Antecedents i estat de l'art	30	20	600
Estudis comparatius i adquisició	30	20	600
Instal·lació de tots els components	30	30	900
Desenvolupament Front-End	30	70	2.100
Desenvolupament Back-End	30	110	3.300
Desenvolupament inicial Microcontrolador	30	60	1.800
Electrònica i desenvolupament definitiu	30	20	600
Memòria	30	30	900
Total		360	10.800

Les hores inclouen tan el desenvolupament com les hores d'estudi i aprenentatge.

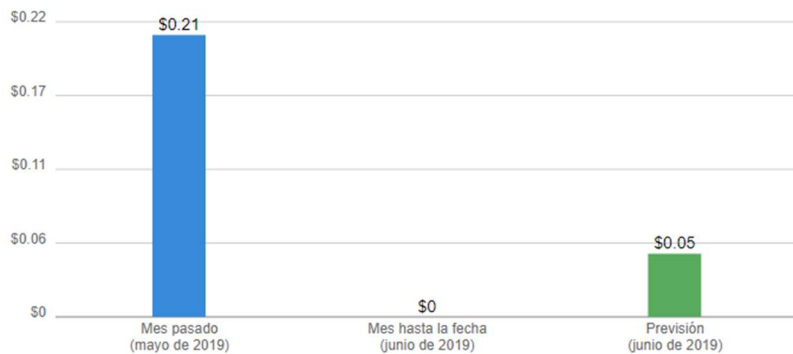
5.2. *Cost del prototip*

El cost del prototip tindrà moltes diferències amb un producte final realment distribuïble, pel tema de la carcassa o els molts canvis que hi ha d'haver per a ser un producte final.

	Preu (€)	Quant.	Import (€)
Microcontrolador Lolin V3	5,20	3	15,60
Sensor d'intensitat SCT-013-000	11,99	1	11,99
Relé 3V amb optoacobrador	3,36	5	16,81
Cable Schuko	8,10	1	8,10
Total	28,65		52,50

El cost unitari és el que cal prendre com a referència.

5.3. Cost del servei



► Información importante acerca de estos costos

Principales servicios de capa gratuita por uso			Ver todos
Servicio	Limite de uso de capa gratuita	Uso mensual hasta la fecha	
Amazon Elastic Compute Cloud	30 GB of Amazon Elastic Block Storage in any combination of General Purpose (SSD) or Magnetic	54.03% (16.21/30 GB-Mo)	
Amazon Relational Database Service	750 hours of Amazon RDS Single-AZ db.t2.micro Instances	53.20% (399.00/750 Hrs)	
Amazon Elastic Compute Cloud	750 hours of Amazon EC2 Microsoft Windows Server† t2.micro instance usage	52.13% (391.00/750 Hrs)	
AmazonCloudWatch	5 GB of Log Data Ingestion for Amazon Cloudwatch	3.00% (0.15/5 GB)	
AmazonCloudWatch	5 GB of Log Data Archive for Amazon Cloudwatch	0.58% (0.03/5 GB-Mo)	

Alertas y notificaciones

- 📌 Cumple los requisitos para la [capa de uso gratuito de AWS](#). Consulte la [Guía de introducción de la capa de uso gratuito de AWS](#) para saber cómo empezar a usar la capa de uso gratuito.
- 📌 Presupuestos de AWS permite crear presupuestos de costo y uso personalizados que le avisan cuando

Com ja s'ha comentat els costos del servei a AWS són complexos.

La factura de Maig va ser de \$0,21 i la de Juny preveuen que sigui de \$0,05. Això és degut a que els nivells de transferència, volum de dades, hores d'us i registre d'events està per sota del llindar de cobrament com es pot veure en la figura anterior.

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 33 of 35		

6. TREBALL FUTUR

Per finalitzar el projecte global caldria desenvolupar projectes en les següents àrees:

6.1. *Dispositius de la llar*

A partir del prototipus desenvolupat en aquest projecte, o d'altres que siguin compatibles, cal desenvolupar productes més compactes, que siguin de molt senzilla implantació, i ajustats al màxim al preu.

Caldria definir un conjunt més complert, com interruptors, sensors de presència, de temperatura o de llum, i si els estàndards ho permeten s'hi hauria d'incloure informació dels electrodomèstics i plaques solars. També millorar la qualitat del sensor de corrent.

La majoria d'aquestes coses ja han estat assolides per altres companyies que s'han esmentat al capítol 2.5 i altres que no s'han inclòs. Seria bo plantejar-se establir col·laboracions i analitzar amb més profunditat el projecte de OpenEnergyMonitor.

6.2. *Gestió de l'energia*

En aquest projecte només es fa l'enregistrament de les dades de corrent, i un simple algorisme de gestió que només té per objectiu demostrar-ne la viabilitat, però que no té cap valor real per a l'estalvi d'energia.

Caldria definir informes més útils a nivell de gestió individualitzada del consum i establir els indicadors més bàsics.

6.3. *Benchmarking*

Aquesta és la part que dona més sentit a aquest projecte però no es podrà dur a terme sense tenir les parts 6.1 i 6.2 molt avançades.

La complexitat en aquest apartat vindrà molt més de la part "comercial" en el sentit de convèncer a grups d'usuari de treballar en equip per a millorar l'eficiència energètica no de l'edificació o dels electrodomèstics, sinó també de l'ús.

El mètode de treball i el que cal fer en aquesta fase s'ha explicat molt resumidament en el capítol 2.1

6.4. *Automatització de la millora*

Finalment i en paral·lel al que es pugui estar fent a nivell de benchmarking, caldria treballar sobre les dades aconseguides de forma automatitzada amb tècniques de I.A., Machine Learning i la resta esmentades en el capítol 2.6.

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 34 of 35		

7. CONCLUSIONS

El projecte ha aconseguit els seus objectius, doncs s'ha aconseguit connectar un element de la llar amb un sistema centralitzat amb les següents característiques:

- a) Presa periòdica d'informació d'un sensor.
- b) Portar un registre centralitzat de la informació del sensor.
- c) Accionar un actuator (relé) des d'un dispositiu distant a voluntat d'un usuari.
- d) Establert un algorisme (alarma) que accioni un actuator en la llar sense participació directa de l'usuari, però a partir d'un llindar introduït per ell.

I aquests objectius s'han aconseguit:

- e) Amb una arquitectura que permet un creixement pràcticament il·limitat.
- f) I amb uns costos dels components de la llar que indiquen una viabilitat futura.

A més, la implementació d'aquest prototipus ha permès veure alguns problemes o dificultats que caldrà resoldre si es decideix portar el projecte fins al final, que són:

- a) Resoldre els problemes de precisió en la lectura del corrent.
- b) Obtenir uns equips senzills, compactes, de baix cost i que siguin pràctics.
- c) Simplificar el procés d'instal·lació i personalització d'aquests dispositius i/o altres de la mateixa família.

Així doncs per la part de software no es veu que hi pugui haver cap limitació, però per la part de l'habitatge, cal treballar-ho molt encara.

Pel que respecte a l'interès acadèmic, el projecte ha estat interessant perquè ha calgut posar en pràctica i en alguns cassos adquirir coneixements en àrees tan diverses com:

- a) Electrònica. Pel fet de dissenyar i construir físicament el prototip amb components reals.
- b) Programació en Arduino i el seu entorn de desenvolupament propi i la seva estructura de programació característica basada en un setup i un bucle.
- c) Programació en Javascript i en l'entorn Node.js i editor Sublime, basat en events i comunicació asíncrona per sockets.
- d) Html i css dins el mateix IDE que Node.js, però com a llenguatges diferent.
- e) SQL per a les consultes i actualitzacions de la base de dades.
- f) MQTT, un sistema de missatgeria de publicació i subscripció i l'eina MQTTLens per a seguir-ne la traça.

Document: Memoria TFG PS	Project Proposal and Work Plan IoT en smart home	 
Date: 25-jun.-19		
Rev: 00		
Page 35 of 35		

8. BIBLIOGRAFIA

- [1] Robert C. Camp. "Benchmarking". Primera edición ; 1993. Editorial Panorama Editorial, S.A.
- [2] Industria 4.0 [online] https://es.wikipedia.org/wiki/Industria_4.0 [accedit Juny 2019].
- [3] Internet de las cosas [online] https://es.wikipedia.org/wiki/Internet_de_las_cosas [accedit Juny 2019].
- [4] PickData, 2019. <https://www.pickdata.net/> [accedit Juny 2019].
- [5] "The home of Energy Monitoring", Efergy Technologies SL, 2018 [online] <https://efergy.com/> [Accedit: Juny 2019].
- [6] WattioCorp S.L., [online] <https://wattio.com> [Accedit: Juny 2019].
- [7] OpenEnergyMonitor [online] <https://openenergymonitor.org/> [accedit Juny 2019].
- [8] "MQTT ESSENTIALS" HiveMQ [online] <https://www.hivemq.com/tags/mqtt-essentials/> [accedit Juny 2019].
- [9] "Getting Started with Arduino and Genuino products". Arduino, 2019. [online] <https://www.arduino.cc/en/Guide/HomePage> [accedit Juny 2019].
- [10] Node.js Foundation. Joyent, Inc [online] <https://nodejs.org/es/> [accedit Juny 2019].
- [11] *Institut Català d'Energia*. [online] <http://icaen.gencat.cat/ca/inici> [accedit Juny 2019].